# Path Planning for Grasping Operations Using an Adaptive PCA-based Sampling Method

Jan Rosell, Luis Cruz, Raúl Suárez and Alexander Pérez

***Abstract*— The planning of collision-free paths for a hand-arm robotic system is a difficult issue due to the large number of degrees of freedom involved and the cluttered environment usually encountered near the grasping configurations. To cope with this problem, this paper presents a novel importance sampling method based on the use of Principal Component Analysis to enlarge the probability of finding collision-free samples in these difficult regions of the Configuration Space with low clearance. By using collision-free samples near the goal, PCA is periodically applied and used to obtain a sampling volume near the goal that better covers the free space, improving the efficiency of sampling-based path planning methods. The approach has been tested with success on a hand-arm robotic system composed of a four-finger anthropomorphic mechanical hand (17 joints with 13 independent degrees of freedom) and an industrial robot (6 independent degrees of freedom).**

***Index Terms*— Importance sampling, Principal Component Analysis, Anthropomorphic hands, Motion planning.**

## I. Introduction

Robotics is continually broadening its field of application, mainly towards service robotics, following advances in all of its disciplines. The improvement of manipulation capabilities is decisively contributing to this tendency. To this end hand-arm robotic systems are being developed not solely within the scope of humanoid robotics but also for mobile manipulators. There are anthropomorphic mechanical hands with a number of DOF ranging from 12 (four fingers with 3 independent DOF each one) to 25 (five fingers with 4 independent DOF each one plus some DOF in the palm) [1], [2]. Therefore, hand-arm robotic systems are complex mechanisms with many degrees of freedom, and the automatic determination of their movements is difficult due to the high dimensionality of the corresponding Configuration Space ($\mathcal{C}$space).

To cope with high-dimensional path planning problems, sampling-based approaches have been proposed. These methods avoid the explicit characterization of the $\mathcal{C}$space, requiring only the collision evaluation of a discrete set of sample configurations and their interconnection with simple collision-free paths [3]. Despite its simplicity, these methods have successfully solved many difficult problems involving a large number of degrees of freedom, being its efficiency tied to the capability of sampling those regions of the $\mathcal{C}$space relevant to the query to be solved, i.e. the sampling procedure is a key factor of this kind of planners. Different importance sampling strategies have been proposed towards this end [4], [5], like those that over-sample the $\mathcal{C}$space but quickly filter any not-promising configuration (e.g. [6], [7]), or those that bias the sampling using the information gathered during the construction of the roadmap or tree (e.g. [8], [9]). In order to improve the performance of sampling-based planners, dimension-reduction techniques have also been proposed, e.g. by using information provided by the user or by the constraints of the task [10], [11], or by capturing the coupling that there may exist between the degrees of freedom of the mechanism using Principal Component Analysis (PCA) [12], [13].

PCA has also been used to bias sampling, as first proposed in [14] to accelerate the diffusion of a Rapidly-exploring Random Tree (RRT) within a narrow passage. That approach modified the traditional extension step of the RRT algorithm by applying the PCA to a set of neighbors of the node to be extended and by changing the extending direction according to the PCA-obtained directions in which the variance of the growing tree is high. The approach does not solve the problem of finding the entrance of a narrow passage, but only accelerate the diffusion within the passage. Another work [15] proposed the use of PCA, within the scope of a Probabilistic RoadMap planner (PRM), to determine a region that tightly bounds the free space of a difficult area of the $\mathcal{C}$space (e.g. a narrow passage), that is used as a sampling region. The approach requires the specification of the region where the narrow passage might approximately be. As a difference with [14], the approach in [15] does not require a PCA computation per sample. Based on the approach of [15], the present paper proposes a PCA-based PRM for a hand-arm robotic systems focused on the grasping configurations, where the environment is cluttered and the solution paths have low clearances. Besides the use of PCA to bias samples, the planner proposed here also uses the PCA as a dimension-reduction technique to obtain human-like motions [16].

The paper is structured as follows. Sections II and III deal, respectively, with the Principal Component Analysis method as a dimension-reduction technique and as a an importance sampling method, and Section IV proposes a planner that incorporates both approaches for the planning of a hand-arm robotic system composed by an industrial robot and an anthropomorphic mechanical hand. Finally, Section V presents the conclusions of the work.

This work was partially supported by the Spanish Government through the projects PI09/90088, DPI2008-02448 and DPI2010-15446.

The authors are with the Institute of Industrial and Control Engineering (IOC) - Technical University of Catalonia (UPC), Barcelona, Spain `jan.rosell@upc.edu`. A. Pérez is also with the Escuela Colombiana de Ingeniería "Julio Garavito", Bogotá D.C.

## II. Principal Component Analysis

Principal Component Analysis (PCA) is a statistical technique used to process a set of vectorial samples in order to look for a new orthogonal base of the vectorial space whose axis indicate, in a decreasing order, the directions of the space with more information to discriminate the samples, i.e. the dispersion of the samples is maximal along the first direction of the new base and decreases along the remaining ones. PCA is a common preprocessing step used to simplify the problem in pattern recognition and classification applications as well as in compression schemes and, in the field of motion and path planning, it is frequently used to reduce the dimension of the searching space and therefore decrease the running time of the planning procedures.

There are different ways of performing the PCA [17]. Basically, it can be done by computing the eigenvalue decomposition of a data covariance matrix or the singular value decomposition of a data matrix, usually after mean centering the data for each attribute. The larger the eigenvalues or the singular values the larger the dispersion of the data along the corresponding eigenvector direction; the eigenvectors are directly used to define the directions of the new base.

### A. Dimension reduction using PCA

PCA is frequently used to reduce the dimension $n$ of the initial working space, using instead a subspace of dimension $m < n$ defined by the first $m$ directions of the new base obtained with the PCA and neglecting the others. Figure 1 shows a simple illustrative example of the use of PCA to reduce the dimension of the working space $\mathcal{C}$. The grey dots represent samples $\vec{x}$ in a 2-dimensional space defined by the original variables $x_1$ and $x_2$ (which may represent two real features of the problem). $O'$ represents the mean of the set of samples, so the samples are first modified as $\vec{x}' = \vec{x} - O'$. Then, using PCA, a new base defined by $x'_1$ and $x'_2$ (which could be considered 2 virtual features) determines a new reference system with the origin at $O'$. Now, since the dispersion of the samples is larger along $x'_1$, the component $x'_2$ is neglected, which is equivalent to consider the subspace $\mathcal{SC} \subset \mathcal{C}$ defined only by $x'_1$ as the working space instead of $\mathcal{C}$, so the dimension of the working space was reduced from 2 to 1. Finally, the actual workspace for the generation of new samples is constrained to a portion $V_S$ of $\mathcal{SC}$ defined by the range $[-\lambda_1, \lambda_1]$, such that it includes a desired percentage of the original samples.

### B. Principal Motion Directions

The directions determined by the axis $x'_i$ of the base obtained with the PCA were also used to define a set of corresponding motion directions in the workspace $\mathcal{C}$ that were called Principal Motion Directions (PMD) [13]. PMDs are ordered following the same criterion that the axis $x'_i$ of the new base, therefore a motion along the first PMDs has a larger range of motion in $V_S$ than along the second one, and so on until the last one. Then, starting from any sample in $\mathcal{C}$ allowing movements along the first $m$ PMDs will likely allow to cover a significant portion of the valid workspace.

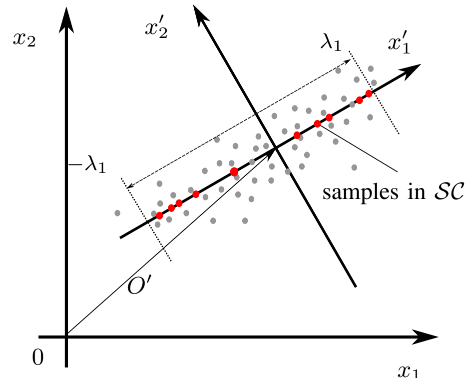


Fig. 1: Illustration of the use of PCA to reduce the dimension of the workspace. The original 2-dimensional space $\mathcal{C}$ is defined by $x_1$ and $x_2$, the mean of the set of samples is $O'$, the direction with maximal dispersion of samples is given by $x'_1$, and the new base is defined by $x'_1$ and $x'_2$. Then, the new working subspace is defined by $x'_1$, and the valid portion $V_S$ is constrained to the range $[-\lambda_1, \lambda_1]$.

### C. Path planning using PMDs

The PMDs were used in the problem of path planning applied to a hand-arm system [13]. The main steps of this approach are the following:

1) A human operator wears a sensorized glove (Cyberglove with 22 sensors) that allows the identification of the hand configurations. Then, while making free natural movements of the hand in a free space a set of samples are captured, which represent the natural workspace of the operator's hand. Each sample is a vector with 22 components.
2) The obtained samples are mapped to the configuration space of a mechanical hand so as to make the mechanical hand postures look as much as possible as those of the operator hand (the Schunk Anthropomorphic Hand SAH is used, it has 17 joints but only 13 degrees of freedom because the distal and medium joints of each finger are coupled).
3) Once the samples are mapped into the mechanical hand configuration space (with dimension 13) a Principal component Analysis is performed and a new base of the space is obtained defining the corresponding PMDs.
4) Applying the idea of dimensionality reduction, only a reduced number of PMDs is used (3 PMDs covers almost 85% of the total variance of the analyzed dataset) to generate samples in the mechanical hand joint space in order to look for a solution path using a PRM, which also considers the 6 degrees of freedom of the robot arm. Since the initial and goal configurations may be outside the reduced workspace of the hand these are specifically connected to the PRM. The consideration of a variable (increasing) number of PMDs in the planning phase was also explored [16], [18].

The approach proposed in this work follow this line, but besides the reduction of the search space, it proposes a new

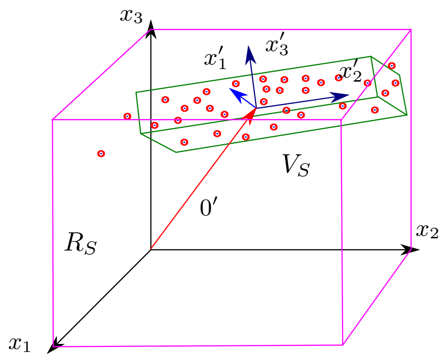


Fig. 2: Region $R_S$ and sampling volume $V_S$ obtained using PCA for a three dimensional $\mathcal{C}$space.

original use of the PCA to improve the sampling phase for the application of the PRM.

## III. Importance Sampling based on Adaptive Principal Component Analysis

### A. The key idea

The key idea of the use of PCA as an importance sampling method for a PRM is twofold: a) the use of PCA to define a new basis for the sampling space able to generate with a greater probability collision-free configurations in difficult areas of the $\mathcal{C}$space (the whole new base is considered, i.e. no reduction of dimensionality is pursued); and b) the periodic recomputation of this basis as new collision-free configurations are obtained, in order to make the process adaptive and obtain a continuous improvement of the sampling performance. The sampling procedure proposed is conceived as a local method, i.e. it is to be applied to a region of the $\mathcal{C}$space where the area of interest is known to be located (e.g. a narrow passage).

Let $R_S$ and $V_S$ be two regions defined as follows (Fig. 2):

- $R_S$: Region of the $\mathcal{C}$space where importance sampling is required.
- $V_S$: Hyper-box that results from applying PCA to the collision-free samples of $R_S$.

Given a set of collision-free samples of $R_S$, the Principal Component Analysis is used to obtain $V_S$. Then, the sampling procedure samples configurations from both regions $R_S$ and $V_S$ and stores them to update $V_S$ in the next call to the algorithm (the reason behind the sampling in $R_S$ is the obtention of collision-free configurations not included in $V_S$ that allow the recomputation of $V_S$ to better cover the area of interest).

### B. Procedure

Algorithm 1 describes a PCA-based sampling procedure. This is a variation of the algorithm proposed in [15] that is simpler and guarantees the obtaining of collision-free configurations from both $V_S$ and $R_S$. It uses the following functions:

- SAMPLE-FROM($B$, $n$): Returns $n$ collision-free configurations sampled from region $B$.

**Algorithm 1** PCA-based Sampling

**Require:**
$R_S$: region of $\mathcal{C}$space
$S$: set of at least $d$ collision-free samples from $R_S$
$k$: even number of collision-free configurations to sample
**Ensure:**
$S$ enlarged with up to $k$ new collision-free samples

$V_S$=PCA($S$)
$S_R =$ SAMPLE-FROM($R_S$, $k/2$)
$S = S \cup S_R$
**for** $i = 1$ to $k/2$ **do**
  $s =$ SAMPLE-FROM($V_S$, 1)
  **if** $s \in R_S$ **then**
    $S = S \cup s$
  **end if**
**end for**
**return** $S$

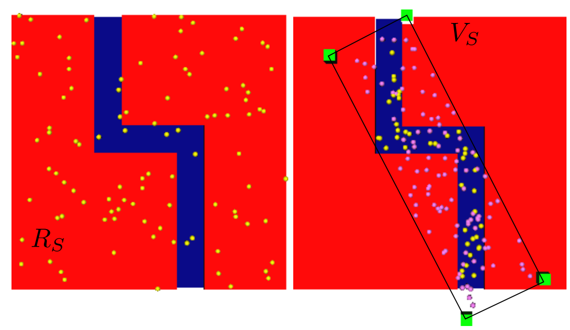


Fig. 3: Examples of samples obtained from regions $R_S$ and $V_S$ for an S-shaped narrow passage.

- PCA($S$): Performs the Principal Component Analysis over the samples of the set $S$ and returns an hyper-box aligned with the resulting new base, centered at the mean value of $S$, and with the length of each side equal to three times the standard deviation of the data along the corresponding axis.

Given an initial set $S$ of at least $d$ collision-free configurations (being $d$ the dimension of the $\mathcal{C}$space) and given a region $R_S$, the sampling procedure computes $V_S$ using the Principal Component Analysis and then samples $k$ collision-free configuration from both $R_S$ and $V_S$. The algorithm returns the set $S$ enlarged with the new collision-free configurations that belong to $R_S$ (take into account that $V_S$ may not be completely inside $R_S$ and therefore some configurations sampled from $V_S$ may not pertain to $R_S$). Fig. 3 shows two snapshots to illustrate the sampling procedure.

## IV. Application to the Path Planning of a Hand-Arm System

This section proposes a PRM planner that uses the PCA-based sampling method to solve path planning problems of a hand-arm robotic system. The proposed PRM is conceived as a single-query problem, i.e. it is not constructed to capture the connectivity of the whole free space of the $\mathcal{C}$space, but solely the part that is relevant to connect two given configurations, $c_{ini}$ and $c_{goal}$, being the later a grasp configuration. In the following subsections, the $\mathcal{C}$space of the problem and the regions where samples are to be obtained are defined; then the PRM is described and evaluated.

### A. The Configuration Space of the problem

Let $\mathcal{C}$ be the $\mathcal{C}$space of a hand-arm robotic system:

$$\mathcal{C} = \mathcal{C}^{\text{a}} \times \mathcal{C}^{\text{h}} \tag{1}$$

where $\mathcal{C}^{\text{a}}$ and $\mathcal{C}^{\text{h}}$ are, respectively, the $\mathcal{C}$spaces of the arm and of the mechanical hand. Using the Principal Motion Directions described in Section II-B, the path planning will be done in a subspace $\mathcal{SC}$ defined as:

$$\mathcal{SC} = \mathcal{C}^{\text{a}} \times \mathcal{SC}^{\text{h}} \tag{2}$$

where $\mathcal{SC}^{\text{h}}$ is the $H$-dimensional subspace of $\mathcal{C}^{\text{h}}$ defined by the first $H$ PMDs. Then, if $A$ is the dimension of $\mathcal{C}^{\text{a}}$, the planning will be done in a $d$-dimensional space with:

$$d = A + H.$$

Therefore, a configuration $c \in \mathcal{SC}$ will be a $d$-dimensional vector whose first $A$ components correspond to the joints of the arm and whose last $H$ components correspond to the $H$ PMDs considered, and that determine the joints of the hand.

### B. Sampling regions

This subsection defines the subregions of $\mathcal{SC}$ where to obtain samples for the proposed PRM. Let:

- $\text{pos}(c)$ be the function that returns the position coordinates of the arm when the hand-arm system is located at configuration $c$,
- $\text{dist}(p_1, p_2)$ be the function that computes the Euclidean distance between two points $p_1, p_2 \in \mathbb{R}^3$, and
- $B$ be a region of $\mathcal{SC}$ defined as:

  $$B(p, \delta) = \{c \in \mathcal{SC} \mid \text{dist}(\text{pos}(c), p) \leq \delta\}, \tag{3}$$

  with $p \in \mathbb{R}^3$ and $\delta$ a given distance threshold.

Then, the following regions are defined:

- Region $R_S$: Usually, the paths of a hand-arm system to grasp an object have low clearances near the goal grasp configuration. Therefore, it is near the goal configuration where the PCA-based sampling proposal better contributes to improve sampling-based planners. For this reason, $R_S$ is defined as the region of $\mathcal{SC}$ satisfying that the position of the arm (i.e. the $x$, $y$ and $z$ coordinates of the robot wrist) is at most at a given distance $\delta_R$ from the position of the arm when it is located at the goal grasp configuration $c_{goal}$:

  $$R_S(\delta_R) = B(\text{pos}(c_{goal}), \delta_R) \tag{4}$$

  The value of $\delta_R$ is approximately chosen equal to the finger lengths, and therefore it is set to 15 mm for the SAH hand.

  The Principal Component Analysis uses the collision-free configurations in $R_S$ to recompute the sampling region $V_S$.
- Region $I_S$: In order to compute $V_S$ for the first time, a set of at least $d$ samples are required (i.e. a number equal to the dimension of the $\mathcal{C}$space). These are obtained by sampling around $c_{goal}$, i.e. by obtaining samples from the region $I_S$ defined as:

  $$I_S(\delta_I) = B(\text{pos}(c_{goal}), \delta_I), \tag{5}$$

  with $\delta_I$ being a threshold smaller than $\delta_R$ (set to 5 mm for the SAH hand).

  Sampling from $I_S$ is done as follows:
  - Arm configuration: The arm position is obtained by sampling a cube of side $\delta_I$ centered at $\text{pos}(c_{goal})$; the arm orientation is obtained by sampling around the goal orientation (the orientation is parameterized with three parameters using quaternions [19], and these parameters are varied an small amount around the values corresponding to $c_{goal}$). Afterwards the arm configuration $(\theta_1, \ldots, \theta_A) \in \mathcal{C}^{\text{a}}$ is obtained using the inverse kinematics (the same inverse kinematics configuration as $c_{goal}$ is chosen).
  - Hand configuration: The hand configuration $(\theta_{A+1}, \ldots, \theta_{A+H}) \in \mathcal{SC}^{\text{h}}$ is obtained by sampling each PMD around the values corresponding to $c_{goal}$.
- Region $V_S$: It is the region resulting from the Principal Component Analysis applied to all collision-free hand-arm configurations within $R_S$. It is a hyper-box of dimension $d$. Sampling within $V_S$ results in hand-arm configurations $(\theta_1, \ldots, \theta_A, \ldots, \theta_{A+H}) \in \mathcal{SC}$.
- Region $M_S$: Samples further away from $c_{goal}$ are needed to construct the whole roadmap to solve the query to connect $c_{ini}$ and $c_{goal}$. Also, collision-free samples outside $V_S$ are needed to recompute $V_S$ to improve the coverage of the area of interest. For these purposes, a region $M_S$ is defined in a similar way as $R_S$ and $I_S$ but with a variable distance threshold that ranges from $\delta_I$ to the distance between the initial and the goal arm positions, i.e.:

  $$M_S(\delta_M) = B(\text{pos}(c_{goal}), \delta_M) \tag{6}$$

  with $\delta_I \leq \delta_M \leq \text{dist}(\text{pos}(c_{ini}), \text{pos}(c_{goal}))$.

  Sampling from $M_S$ is done as follows:
  - Arm configuration: The arm position is obtained as done in $I_S$ but using $\delta_M$ instead of $\delta_I$.
  - Hand configuration: The hand configuration is obtained by sampling each PMD within its whole range.

### C. PCA-based PRM

Algorithm 2 describes a Probabilistic RoadMap that uses the PCA-based sampling method to solve problems for a real hand-arm robotic system; it is called PCA-based PRM. It uses the functions of Section III-B plus the following:

- PRM-SOLVE($s$): Adds the collision-free configuration $s$ to the roadmap and returns the path connecting the initial to the goal configuration, if it exists, or the empty-set otherwise.

**Algorithm 2** PCA-based PRM

**Require:**
$c_{ini}$: Initial configuration
$c_{goal}$: Goal configuration
$d$: Dimension of the $\mathcal{C}$space
$\delta_I$, $\delta_R$: Distance thresholds
$N$: Number of samples per iteration (an even value)
**Ensure:**
$P$: path connecting $c_{ini}$ and $c_{goal}$

```
S_PCA = SAMPLE-FROM(I_S, d)
PRM-ADD(S_PCA)
δ_M = δ_I
trials = 0
while trials < MaxTrials do
   trials = trials + N
   V_S=PCA(S_PCA)
   for i = 1 to N/2 do
      s = SAMPLE-FROM(V_S, 1)
      P=PRM-SOLVE(s)
      if P ≠ ∅ then
         return P
      end if
      if s ∈R_S then
         S_PCA = S_PCA ∪ s
      end if
   end for
   for i = 1 to N/2 do
      s = SAMPLE-FROM(M_S(δ_M), 1)
      P=PRM-SOLVE(s)
      if P ≠ ∅ then
         return P
      end if
      if s ∈R_S then
         S_PCA = S_PCA ∪ s
      end if
   end for
   δ_M = δ_M + δ_I
   if δ_M > dist(pos(c_ini), pos(c_goal)) then
      δ_M = δ_I
   end if
end while
return ∅
```

The algorithm first obtains collision-free configurations around the goal configuration by sampling within $I_S$, then it loops executing the following three steps until the solution is found or a maximum number of samples have been generated:

1) (Re)compute $V_S$ using those samples of the roadmap within region $R_S$.
2) Sample from $V_S$, add the samples to the roadmap and search for a solution.
3) Sample from $M_S$, add the samples to the roadmap and search for a solution.

The size of $M_S$ is not fixed but it is defined with a variable value of $\delta_M$: the initial value of $\delta_M$ equals $\delta_I$, and it is incremented also by $\delta_I$ at each iteration, restarting at $\delta_I$ whenever $\delta_M$ becomes greater than dist(pos($c_{ini}$), pos($c_{goal}$)).

The distance thresholds $\delta_R$ and $\delta_I$ depend on the mechanical hand used. Therefore, the planner has only one extra parameter to be defined by the user (the number of samples per iteration, $N$), besides the parameters of the basic PRM as the number maximum of neighbors per node and the neighbor distance threshold. The value of $N$ is not critical and good results are obtained with $N$ ranging from 20 to 100 (lower values of $N$ require more calls to the PCA and result in a faster adaptation, although the first guess might fit worse the passage).

### D. Examples

The proposed approach has been implemented as a specialized PRM planner inside the home-developed path planning framework. This tool, called *Kautham*, was developed with the open-source and cross-platform directives in mind [20], and it uses libraries such as Qt [21] for the user interface, Coin3D [22] for the graphical rendering and PQP [23] for the collision detection. This application provides the developer with direct and inverse kinematic models of the robots, and with samplers, metrics and other tools needed for the development of planners.

With respect to the PCA introduced in the paper, there are several alternatives for its implementation. One possibility is to use Octave [24] or R [25] together with a package like the RCPP [26] to connect them to the application. However, since our application requires to perform PCA within the sampling loop, the performance criteria is considered a key factor, and for this reason the Armadillo C++ Linear Algebra Library has been used [27]. This library is also open-source and has a good performance in response time for large volumes of data, as reported in [15].

The following two examples illustrate the ability of the proposed approach to find the path for a hand-arm robotic system towards a low-clearance grasp configuration. The hand-arm robotic system is composed of a TX90 Stäubli robot and the Schunk Anthropomorphic Hand, SAH. In the first example the system grasps a T-shaped object (Fig. 4), in the second it grasps a can (Fig. 5). The proposed planner successfully solved the examples (see the accompanying video), although both of them are difficult due to the narrow corridor in $\mathcal{C}$space generated by the fingers and the object to be grasped.

The the number of samples per iteration has been set to $N = 30$. As mentioned in previous sections, the distance thresholds for the SAH hand are set to $\delta_R = 150$ mm and $\delta_I = 50$ mm. Then, setting the maximum number of samples to 4000, the success rate obtained for each example has been 72% and 70%, respectively, the number of nodes $58 \pm 12$ and $49 \pm 15$, and the number of collision-checked configurations $1840 \pm 314$ and $950 \pm 246$. In comparison, no solutions were found using the Gaussian sampler [6] with the same maximum number of samples.

## V. Conclusions

Principal Component Analysis has previously been used in the scope of path planning to reduce the dimensionality of the planning space. In this paper it is also used as an importance sampling method (i.e. as a way to enlarge the probability to obtain samples from those difficult regions of

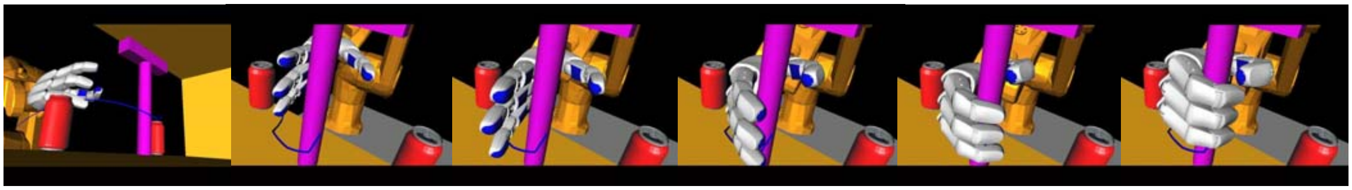

Fig. 4: Planning the grasp of the T shape.

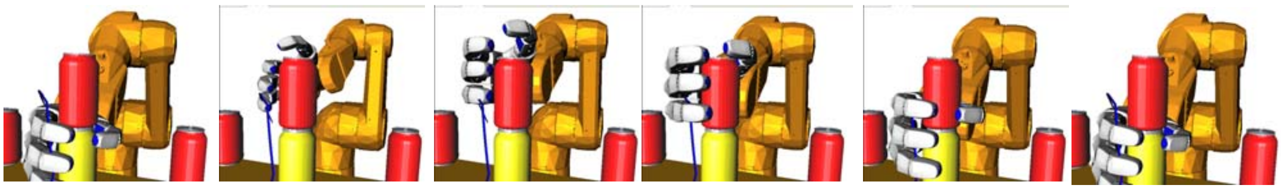

Fig. 5: Planning the grasp of the yellow can.

the Configuration Space relevant to solve a query) within the scope of hand-arm robotic systems. The search of a collision-free path to reach an object to be grasped is a difficult issue due to the low clearances that there exist near the goal grasp configuration, and the large number of degrees of freedom. A probabilistic roadmap path planner that uses the PCA-based sampling method near the goal grasp configuration has been proposed and tested with success on a hand-arm robotic system composed of a four-finger anthropomorphic mechanical hand (17 joints with 13 independent degrees of freedom) and an industrial robot (6 independent degrees of freedom). As a future work we are planning to study the automatic determination of the regions where to apply the proposed sampling method, besides the region around the goal configuration as done in the present paper.

## References


[1] A. Bicchi, "Hands for dexterous manipulation and robust grasping: A difficult road toward simplicity," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 652–662, 2000.

[2] L. Biagiotti, F. Lotti, C. Melchiorri, and G. Vassura, "How far is the human hand? a review on anthropomorphic robotic end-effectors," tech. rep., University of Bologna, 2004.

[3] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion*. The MIT Press, 2005.

[4] R. Geraerts and M. H. Overmars, "Sampling techniques for probabilistic roadmap planners," *Intelligent Autonomous Systems*, vol. 8, pp. 600–609, 2004.

[5] D. Hsu, J.-C. Latombe, and H. Kurniawati, "On the probabilistic foundations of probabilistic roadmap planning," *The Intenational Journal of Robotics Research*, vol. 25, no. 7, pp. 627 – 643, 2006.

[6] V. Boor, M. H. Overmars, and A. F. van der Stappen, "The Gaussian sampling strategy for probabilistic roadmap planners," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 1018–1023, 1999.

[7] D. Hsu, T. Jiang, J. Reif, and Z. Sun, "The bridge test for sampling narrow passages with probabilistic roadmap planners," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 4420–4426, 2003.

[8] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. K. Overmars, "Probabilistic roadmaps for path planning in high - dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, pp. 566–580, August 1996.

[9] D. Hsu, G. Sanchez-Ante, and Z. Sun, "Hybrid PRM sampling with a cost-sensitive adaptive strategy," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 3874 – 3880, 2005.

[10] D. Berenson, S. Srinivasa, D. Ferguson, and J. Kuffner, "Manipulation planning on constraint manifolds," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 625 – 632, 2009.

[11] M. Stilman, "Global manipulation planning in robot joint space with task constraints," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 576–584, 2010.

[12] A. Safonova, J. K. Hodgins, and N. S. Pollard, "Synthesizing physically realistic human motion in low-dimensional, behavior-specific spaces," *ACM Transactions Graph.*, vol. 23, no. 3, pp. 514–521, 2004.

[13] J. Rosell, R. Suárez, C. Rosales, J. A. García, and A. Pérez, "Motion planning for high DOF anthropomorphic hands," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 4025–4030, 2009.

[14] S. Dalibard and J. Laumond, "Control of probabilistic diffusion in motion planning," in *Proc. of the Workshop on Algorithmic Foundations of Robotics*, pp. 467 – 481, 2008.

[15] J. Rosell, L. Cruz, R. Suárez, and A. Pérez, "Importance sampling based on adaptive principal component analysis," in *Proc. of the IEEE Int. Symposium on Assembly and Manufacturing*, 2011.

[16] R. Suárez, J. Rosell, A. Pérez, and C. Rosales, "Efficient search of obstacle-free paths for anthropomorphic hands," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 1773–1778, 2009.

[17] I. Jolliffe, *Principal Component Analysis*. Upper Saddle River, NJ, USA: Springer Series in Statistics, 2002.

[18] J. Rosell, R. Suárez, C. Rosales, and A. Pérez, "Autonomous motion planning of a hand-arm robotic system based on captured human-like hand postures," *Autonomous Robots*, vol. 31, no. 1, pp. 87–102, 2011.

[19] J. Kuffner, "Effective sampling and distance metrics for 3d rigid body path planning," in *Proc. of Int. Conf. on Robotics and Automation*, pp. 3993 – 3998, 2004.

[20] A. Pérez and J. Rosell, "A Roadmap to Robot Motion Planning Software Development," *Computer Applications in Engineering Education*, September 2009.

[21] J. Blanchette and M. Summerfield, *C++ GUI Programming with Qt 4*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2006.

[22] Kongsberg Oil & Gas Technologies, "Coin3D - 3d graphics development tools." www.coin3d.org, December 2010.

[23] E. Larsen, S. Gottschalk, M. C. Lin, and D. Manocha, "Fast proximity queries with swept sphere volumes," in *Proc. of Int. Conf. on Robotics and Automation*, pp. 3719–3726, 2000.

[24] J. W. Eaton, *GNU Octave Manual*. Network Theory Limited, 2002.

[25] T. R. Foundation, "The R project for statistical computing." www.r-project.org, February 2011.

[26] D. Eddelbuettel, "Rcpp: Seamless R and C++ integration." dirk.eddelbuettel.com/code/rcpp.html, February 2011.

[27] C. Sanderson, "Armadillo: An open source c++ linear algebra library for fast prototyping and computationally intensive experiments," tech. rep., NICTA, 2010.